Executive Summary

Lyrasis VIVO Web Assessment

Congratulations!

Your assessment is complete. In May 2021, Deque Subject Matter Experts performed a web accessibility assessment on the Lyrasis VIVO platform based on the Web Content Accessibility Guidelines ("WCAG") Level 2.1 A & AA Standards.

Performing an assessment is a key step in understanding your current state of accessibility. The results in this report will provide you with actionable next steps to improve the accessibility of your website. The recommended fixes will result in a better experience for your users, including people with disabilities. This Executive Summary is a high-level view of the Lyrasis VIVO web assessment results, or you can dive into the full results.

Key Findings

- 457 total issues identified for the 18 pages & components that were assessed.
- Good News! We found no issues that are total blockers for users with disabilities and 51 of the 79 Deque checkpoints did not detect any issues.
- 70% of the total issues found are the types of issues developers can identify, fix, and verify on their own using automated accessibility testing tools like the axe DevTools browser extension.
- 302 issues were related to color contrast and may be fixed with a change in the style sheet.

What is the user impact?

Impact	Impact Description	Total Issues
Blocker	Prevents some users with disabilities from using your core content.	0
Critical	Prevents some users with disabilities from accessing certain parts of your content, potentially rendering it unusable.	42
Serious	Presents serious barriers for some users with disabilities and will partially prevent them from using portions of your content.	396
Moderate	Presents some barriers for users with disabilities that will reduce their overall experience with your content.	9
Minor	Causes some nuisance or can be annoying, but not presenting barriers for users with disabilities.	10

How did we arrive at these results?

This assessment was performed on the Windows 10 operating system using the following:

- Chrome web browser
- NVDA screen reader
- Automated testing using axe-core rules
- Manual testing
- Keyboard-only navigation testing

What does this mean for your customers?

Here is an overview of the top critical and serious issues we found.

Critical Issues

In 42 critical issues, people with disabilities cannot access certain parts of your content, potentially rendering it unusable. Here are the key drivers:

- Action cannot be performed by keyboard alone
  - Many users rely on a keyboard alone because they are not able to use a mouse due to vision or motor disabilities. Content that can be operated with a mouse must also be made operable with a keyboard.
- Custom control invalid
  - Every user interface control must have a role along with any applicable states and properties so that screen reader users know how to interact with the control. If you create a custom control or widget that does not have a native HTML equivalent, you must add the relevant role(s) and any applicable states and properties using ARIA as well as expected keyboard interactions.
- Form elements must have labels
  - When form elements are missing labels, this is problematic for screen reader users because their assistive technology requires labels to read out the intent of each field. When the form labels are missing the screen reader user will not know the expected value to input into the form field.

Serious Issues

396 serious issues were found that may cause serious barriers for your users. Most of the issues we found were in the following three categories:

- Distinguishable/Color Contrast
  - Elements on a web page need to have sufficient color contrast for the user to be able to discern the text from the background. Some people with low vision experience low contrast which makes it hard to distinguish outlines, borders, edges, and details.

- Missing or incorrect name, role, and/or value
  - States and properties are attributes used to convey essential information about an element to screen readers and other assistive technologies. Some roles require certain state and property information - such as the checked/unchecked state of a checkbox. This code needs to be valid in order for a screen reader to convey the information to a user.
- Keyboard focus order
  - When keyboard focusable components do not receive focus in a logical order, people with mobility impairments, reading disabilities, and low vision are all impacted. A logical focus order makes interaction with content predictable for people who rely on a keyboard to interact with web content.

Where were the issues concentrated?

Pages and components with issues by severity/user impact

Test Units	**Critical**	**Serious**	**Moderate**	**Minor**	**Grand Total**
Add Position to Person Form	5	11			16
Capability Map	3	5	1		9
Capability Map Interaction	5	11	1		17
Co-author Network	2	11	1		14
Department Profile	1	38		1	40
Document Profile	1	20	1	4	26
Footer		3			3
Header	1	3			4
Header After Login	1	2			3
Home Page	1	31	1		33
Index Page		56		1	57
Manage Web Pages	1	5			6
Map of Science	8	19	1		28
People Class group Browse		36			36
Person Profile	5	59	3	4	71
Research Class group Browse		37			37
Search Results	5	43			48
Share URI	3	6			9
Grand Total	**42**	**396**	**9**	**10**	**457**

Where do we go from here – next steps?

Developers and Testers should carefully review the detailed assessment report within axe Auditor as it contains the full results of the assessment. It contains screenshots, code snippets and specific recommendations for fixes. If additional report access is needed you may contact your Deque Project Manager for account setup.

For remediation

For addressing these issues, we suggest considering the following when prioritizing:

- Correct the critical issues as soon as possible.
- Make changes with the greatest ROI such as in the style sheet (color contrast) or in common components.
- You may find that correcting issues by page/component works better for your team.
- Developers should download and use Deque's free axe DevTools browser extension to conduct quick, automated testing of their own.

Beyond the immediate remediation

- Use these findings as an input into efforts to create and maintain a sustainable accessibility program that integrates accessibility testing into development processes. Catching an accessibility defect during development costs around 30 times less than doing so in production and is a great example of the business case for accessibility.
- Consider developing an accessibility policy, if you do not have one, to support accessibility program activities. Internal accessibility ownership as well as measuring, tracking, and reporting on goals are key to being successful and efficient with accessibility in the long run.